

BAB II TINJAUAN PUSTAKA

Pada bagian tinjauan pustaka akan dijelaskan mengenai penelitian terdahulu serta berbagai referensi yang terkait dan menunjang permasalahan yang akan diteliti. Referensi tersebut berkaitan dengan penjadwalan produksi, metode heuristik dan referensi lainnya. Bab ini bertujuan untuk mendukung permasalahan yang akan diteliti serta mendukung hasil penelitian.

2.1 Penelitian Terdahulu

Ada beberapa penelitian sebelumnya yang berkaitan dengan penjadwalan produksi. Berikut penjelasan singkat beberapa penelitian tersebut:

1. Chen (1997), dalam penelitiannya membahas mengenai masalah penjadwalan pada mesin paralel dimana *job* tiba dari waktu ke waktu sehingga waktu prosesnya baru diketahui di kedatangan *job*. Dalam penelitian tersebut, ditunjukkan sebuah algoritma *on-line* LPT memiliki jaminan kinerja sebesar $\frac{3}{2}$, tidak tergantung dari jumlah mesin. Hasil penelitian membuktikan dengan analisa *worst-case*, algoritma ini memiliki kinerja paling tidak pada batas (*lower bound*) 1,3473 dan 1,3820 untuk 2 mesin paralel identik.
2. Koulamas (2009), dalam penelitiannya menunjukkan bahwa kinerja heuristik LPT dapat ditingkatkan untuk masalah 2 mesin paralel identik dalam meminimasi nilai *makespan* (C_{max}). Penelitian ini dilakukan dari *job* $j = 1$ sampai dengan $j = 5$. Hasil pengembangan algoritma LPT yang disebut sebagai *modified LPT* (MLPT) menunjukkan *worst-case ratio* menurun dari 1,27 menjadi 1,2247.
3. Faizal (2014), dalam penelitiannya membahas mengenai masalah penjadwalan pada *re-entrant flowshop* di PT. Sahabat Rubber Indonesia. Dalam penelitiannya, dikembangkan model algoritma Nawaz, Ensore dan Ham (NEH) dengan dua pendekatan *dispatching rule* yaitu SPT dan LPT, dengan memperhatikan *routing* dan *precedence*. Hasil penelitian ini menunjukkan berdasarkan perhitungan *efficiency index* dan *relative error*, model algoritma NEH dengan pendekatan SPT maupun LPT

lebih baik dan menghasilkan penurunan nilai *makespan* sebesar 46,27% dibandingkan penjadwalan *existing*.

4. Wildan (2014), dalam penelitiannya bertujuan untuk menentukan jadwal produksi berdasarkan program produksi bulan Desember 2012 yang memiliki nilai *makespan* optimal. Penjadwalan dilakukan pada 2 mesin pengisian paralel identik di PT. Pertamina Production Unit Gresik. Dalam penelitian ini, dipertimbangkan masalah *sequence dependent setup time* yang dibuat dalam bentuk matriks. Metode yang digunakan adalah *Mixed Integer Linear Programming* dengan *software* LINGO 14.0 serta dihasilkan urutan pengerjaan produk yang optimal untuk masing-masing lintasan dengan nilai *makespan* sebesar 453,5 jam dan nilai utilitas lintasan sebesar 0,91.

Penelitian ini dilakukan pada bagian proses *extrusion* PT. "X" dimana terdapat kapasitas paling rendah yang menyebabkan kecepatan keseluruhan proses produksi dan banyaknya *output* produk tergantung pada proses tersebut. Sebelum dilakukan penjadwalan, dilakukan pemisahan *job* produk berdasarkan warna. Dipertimbangkan pula adanya *job* sisipan dan masalah *sequence dependent setup time* yang akan dibuat dalam matriks waktu *setup*. Kemudian dilakukan pengembangan penjadwalan dengan metode heuristik berbasis LPT untuk mendapatkan urutan *job* yang paling optimal guna memperkecil nilai *makespan*. Selanjutnya, hasil penjadwalan dengan metode tersebut dibandingkan dengan penjadwalan *existing* perusahaan sehingga dapat dipilih penjadwalan paling optimal. Tabel 2.1 merupakan tabel perbandingan penelitian pendahuluan dengan penelitian yang akan dilakukan.

Tabel 2.1 Komparasi Penelitian dengan Penelitian Terdahulu

Peneliti	Objek	Metode	Hasil Penelitian
Chen (1997)	-	Algoritma <i>on-line</i> LPT	Menunjukkan algoritma <i>on-line</i> LPT memiliki kinerja paling tidak pada batas (<i>lower bound</i>) 1,3473 dan 1,3820 untuk 2 mesin paralel identik dengan mempertimbangkan masalah penjadwalan dimana <i>job</i> tiba dari waktu ke waktu.
Koulamas (2009)	-	Modifikasi Algoritma LPT	Menunjukkan kinerja heuristik LPT dapat ditingkat pada penerapan masalah penjadwalan mesin paralel identik dari <i>worst-case ratio</i> 1,27 menjadi 1,2247.
Faizal (2014)	<i>Re-entrant Flowshop</i> di PT. Sahabat Rubber Indonesia	Algoritma NEH dengan pendekatan <i>dispatch rule</i> (SPT, LPT)	Menunjukkan model algoritma NEH dengan pendekatan SPT maupun LPT lebih baik dan menghasilkan penurunan nilai <i>makespan</i> sebesar 46,27% dibandingkan penjadwalan <i>existing</i> berdasarkan perhitungan <i>efficiency index</i> dan <i>relative error</i> .
Wildan (2014)	Mesin Pengisian paralel identik di PT. Pertamina Production	<i>Mixed Integer Linear Programming</i> (MILP)	Menjadwalkan 2 mesin paralel identik dengan <i>sequence dependent setup time</i> (dalam matriks) dan menghasilkan urutan pengerjaan produk yang optimal untuk kedua lintasan dengan nilai <i>makespan</i> terkecil.
Penelitian ini (2015)	Mesin <i>Extruder</i> Paralel Identik di perusahaan tali tampar	Metode heuristik berbasis LPT	Memberikan usulan yang optimal untuk penjadwalan 3 mesin paralel identik dengan mempertimbangkan <i>job</i> sisipan dan masalah <i>sequence dependent setup time</i> (dalam matriks) dalam meminimasi nilai <i>makespan</i> sehingga mengurangi waktu lembur pekerja.

2.2 Penjadwalan Produksi

Penjadwalan didefinisikan sebagai proses pengalokasian sumber daya untuk mengerjakan sekumpulan tugas dalam jangka waktu tertentu dengan 2 arti penting sebagai berikut (Baker, 1974: 2):

1. Penjadwalan merupakan suatu fungsi pengambilan keputusan untuk membuat atau menentukan jadwal.
2. Penjadwalan merupakan suatu teori yang berisi sekumpulan prinsip dasar, model, teknik, dan kesimpulan logis dalam proses pengambilan keputusan yang memberikan pengertian dalam fungsi penjadwalan.

Menurut Pinedo (2012: 2), penjadwalan adalah proses pengambilan keputusan yang memegang peranan yang penting dalam manufaktur dan sistem produksi. Penjadwalan berkaitan dengan pengalokasian sumber daya untuk tugas-tugas selama jangka waktu tertentu dan tujuannya untuk mengoptimalkan satu atau lebih tujuan.

Penjadwalan produksi berfungsi untuk menyelesaikan permasalahan yang berhubungan dengan pertanyaan-pertanyaan, seperti apa produk yang akan diproduksi, berapa banyak produk tersebut diproduksi, serta bagaimana alokasi sumber daya yang dimiliki untuk melakukan tugas-tugas yang dibutuhkan dalam proses produksi.

Dikutip dari Bedworth dan Bailey (1987), beberapa tujuan dari aktivitas penjadwalan adalah sebagai berikut (Ginting, 2009: 2):

1. Meningkatkan penggunaan sumber daya atau mengurangi waktu tunggu sehingga total waktu proses dapat berkurang dan produktivitasnya dapat meningkat.
2. Mengurangi persediaan barang setengah jadi atau mengurangi sejumlah pekerjaan yang menunggu dalam antrian ketika sumber daya yang ada masih mengerjakan tugas yang lain.
3. Mengurangi beberapa keterlambatan pada pekerjaan yang mempunyai batas waktu penyelesaian sehingga akan meminimasi biaya penalti.
4. Membantu pengambilan keputusan mengenai perencanaan kapasitas pabrik dan jenis kapasitas yang dibutuhkan sehingga penambahan biaya yang mahal dapat dihindarkan.

2.2.1 Input Penjadwalan Produksi

Pekerjaan-pekerjaan yang berupa alokasi kapasitas untuk *order-order*, penugasan prioritas *job*, dan pengendalian jadwal produksi membutuhkan informasi terperinci, dimana informasi-informasi tersebut akan menyatakan *input* dari sistem penjadwalan.

Terdapat beberapa hal yang digunakan sebagai variabel dasar dari permasalahan penjadwalan produksi adalah sebagai berikut (Baker, 1974: 6):

1. Semua *job* yang harus dikerjakan pada tiap operasi yang dibutuhkan untuk memproduksi suatu *order* dari konsumen.
2. Lamanya waktu operasi yang dibutuhkan tiap operasi yang dilakukan.
3. Catatan adanya kegiatan pendahulu yang menjadi batasan untuk melakukan kegiatan selanjutnya.
4. Semua sumber daya yang diperlukan untuk tiap pekerjaan (*job*) dan tiap operasi.

2.2.2 Output Penjadwalan Produksi

Untuk memastikan bahwa suatu aliran kerja yang lancar akan melalui tahapan produksi, maka sistem penjadwalan harus membentuk aktivitas-aktivitas *output* sebagai berikut (Ginting, 2009: 9):

1. Pembebanan (*loading*), yang melibatkan penyesuaian kebutuhan kapasitas untuk *order* – *order* yang diterima diperkirakan dengan kapasitas yang tersedia. Pembebanan dilakukan dengan menugaskan *order* – *order* pada fasilitas – fasilitas, operator – operator, dan peralatan tertentu.
2. Pengurutan (*sequencing*), yang merupakan penugasan tentang *order* – *order* mana yang diprioritaskan untuk diproses terlebih dahulu bila suatu fasilitas harus memproses banyak *job*.
3. Prioritas *job* (*dispatching*), yang merupakan prioritas kerja tentang *job* – *job* mana yang diseleksi dan diprioritaskan untuk diproses.
4. Pengendalian kinerja penjadwalan, yang dilakukan dengan meninjau kembali status *order* – *order* pada saat melalui sistem tertentu dan mengatur kembali urutan – urutan.
5. *Up dating* jadwal, yang dilakukan sebagai refleksi kondisi operasi yang terjadi dengan merevisi prioritas – prioritas.

2.2.3 Istilah dan Notasi dalam Penjadwalan Produksi

Istilah – istilah beserta notasi yang digunakan dalam penjadwalan produksi adalah sebagai berikut (Baker & Trietsch, 2009: 11):

1. *Processing time* (p_j) atau waktu proses.
Merupakan waktu yang dibutuhkan untuk menyelesaikan suatu operasi dari *job* j .
2. *Setup time* (s_j) atau waktu *setup*.

Merupakan waktu yang dibutuhkan untuk kegiatan persiapan dan pengaturan sebelum memproses *job j* di suatu mesin.

3. *Release date* (r_j) atau *ready time*.

Merupakan waktu yang menunjukkan dimana *job j* siap untuk diproses/dikerjakan.

4. *Due date* (d_j) atau batas waktu.

Merupakan batas waktu dimana operasi terakhir dari *job j* harus sudah selesai.

5. *Completion time* (C_j) atau waktu penyelesaian.

Merupakan waktu yang dibutuhkan untuk menyelesaikan *job j* mulai dari saat tersedianya *job* sampai *job* tersebut selesai dikerjakan.

6. *Flow time* (F_j) atau waktu alir.

Merupakan waktu yang dihabiskan *job j* dalam sistem. Dalam kata lain merupakan rentang waktu antara *job* tersedia untuk diproses dengan ketika *job* tersebut selesai.

$$F_j = C_j - r_j \quad (2-1)$$

7. *Lateness* (L_j) atau keterlambatan.

Merupakan selisih antara waktu penyelesaian *job j* dengan batas waktunya (*due date*). Keterlambatan dapat dihitung setelah *job j* selesai menjalankan semua proses, dan dapat bernilai negatif dan positif.

$$L_j = C_j - d_j \quad (2-2)$$

8. *Tardiness* (T_j) atau keterlambatan positif.

Merupakan *lateness* yang bernilai positif karena *job j* diselesaikan sebelum batas waktunya (*due date*).

$$T_j = \max(L_j, 0) \quad (2-3)$$

2.2.4 Kriteria Evaluasi dalam Penjadwalan Produksi

Keberhasilan suatu penjadwalan dapat diukur dengan besaran yang melibatkan informasi dari pekerjaan-pekerjaan yang dijadwalkan. Hal tersebut dapat dicapai dengan menentukan kriteria jadwal yang diinginkan. Jika terdapat n pekerjaan yang akan dijadwalkan, maka tingkat keberhasilan dapat dinilai dari besaran-besaran berikut (Baker & Trietsch, 2009: 12):

1. *Maximum completion time* atau *makespan* (C_{\max}).

Merupakan total waktu penyelesaian semua *job* mulai dari urutan pertama yang dikerjakan pada mesin pertama sampai urutan *job* terakhir pada mesin terakhir.

$$C_{\max} = \max_{1 \leq j \leq n} \{C_j\} = \sum p_j \quad (2-4)$$

2. *Total flow time* total waktu alir (F).

$$F = \sum_{j=1}^n F_j \quad (2-5)$$

3. *Total tardiness* atau total keterlambatan positif (T).

$$T = \sum_{j=1}^n T_j \quad (2-6)$$

4. *Maximum flow time* (F_{\max}).

$$F_{\max} = \max_{1 \leq j \leq n} \{F_j\} \quad (2-7)$$

5. *Maximum tardiness* (T_{\max}).

$$T_{\max} = \max_{1 \leq j \leq n} \{T_j\} \quad (2-8)$$

6. *Number of tardy jobs* atau jumlah *job* yang terlambat (U).

$$U = \sum_{j=1}^n \delta(T_j), \text{ dimana } \delta(T_j) = 1 \text{ jika } T_j > 0 \quad (2-9)$$

$$\delta(T_j) = 0 \text{ jika } T_j \leq 0$$

2.2.5 Sequence-Dependent Setup Time

Sequence-dependent setup time (SDST) merupakan salah satu istilah yang masuk dalam kendala atau *constraints* dalam masalah penjadwalan. Menurut Baker & Trietsch (2009: 166) Situasi ini (SDST) muncul ketika waktu *setup* bukan konstan untuk setiap *job* tetapi tergantung pada *job* sebelumnya dalam urutan. Menurut Pinedo (2012: 16), SDST yang dinotasikan dengan s_{jk} menunjukkan waktu *setup* yang tergantung pada urutan (*sequence*) antara *job* j dan k; s_{0k} adalah waktu *setup* untuk *job* k jika merupakan urutan pertama, sedangkan s_{j0} merupakan waktu *setup* setelah *job* j jika merupakan urutan terakhir. Apabila waktu *setup* antara *job* j dan k tergantung mesin, maka dituliskan s_{ijk} . Apabila s_{jk} tidak dicantumkan pada β , maka semua waktu *setup* diasumsikan sama dengan 0 atau *sequence independent* (tidak tergantung urutan), pada kasus dimana waktu *setup* tersebut dapat secara sederhana telah dimasukkan pada waktu proses.

Waktu *setup* yang tergantung dengan urutan (*sequence dependent*) biasanya ditemukan pada fasilitas tunggal yang menghasilkan beberapa jenis produk atau pada mesin serbaguna yang melakukan berbagai macam tugas. Kondisi waktu *setup* seperti ini biasanya ditemukan dalam produksi produk dengan warna yang berbeda dari cat, kekuatan deterjen dan campuran bahan bakar. Pengamatan yang sama berlaku untuk lini perakitan tertentu di mana *retooling*, inspeksi, atau penataan ulang stasiun kerja bisa mewakili aktivitas *setup* (Baker & Trietsch, 2009: 182).

2.3 Klasifikasi Penjadwalan Produksi

Pada umumnya, klasifikasi penjadwalan dapat dibagi menjadi beberapa macam sesuai dengan karakteristiknya, seperti jumlah mesin yang digunakan dalam proses atau aliran prosesnya. Klasifikasi penjadwalan menurut Pinedo (2012: 14) terbagi menjadi:

1. Penjadwalan mesin tunggal (*single machine*)

Klasifikasi penjadwalan dengan terdapat hanya 1 mesin yang tersedia untuk memproses pekerjaan (*job*).

2. Penjadwalan mesin jamak (*parallel machine*)

Klasifikasi penjadwalan dengan terdapat lebih dari 1 mesin yang tersedia untuk memproses pekerjaan (*job*). Penjadwalan paralel dibagi menjadi:

a. Penjadwalan n *job* pada mesin paralel identik (*identical machines parallel*)

Prinsip dalam penjadwalan paralel identik adalah pengalokasian beban ke mesin yang lebih dahulu *idle*/kosong.

b. Penjadwalan n *job* pada mesin paralel non identik

Paralel non-identik dimana setiap mesin mempunyai fungsi yang sama namun waktu proses berbeda. *Flow time* tidak bisa dievaluasi langsung dari waktu proses. Tidak selalu alternatif waktu terpendek dari setiap *job* akan menjadi keputusan alokasi pada mesin.

c. Penjadwalan n *job* pada mesin paralel *unrelated*

Penjadwalan ini merupakan perluasan dari paralel non identik. Terdapat m mesin paralel, dimana mesin i untuk memproses *job* j maka kecepatan mesin adalah v_{ij} .

Beberapa metode yang dapat digunakan untuk penjadwalan mesin paralel antara lain:

a. Aturan LPT (*Longest processing time*) untuk tujuan meminimasi *makespan*.

b. Aturan SPT (*Shortest processing time*), untuk tujuan meminimasi *completion time*.

c. Aturan LRPT (*Longest Remaining Processing Time*) untuk tujuan meminimasi *maximum lateness*.

3. Penjadwalan *flow shop*

Klasifikasi penjadwalan dengan terdapat m mesin serial dimana masing-masing *job* harus diproses di setiap mesin. Semua *job* harus mengikuti rute yang sama. Setelah proses selesai di satu mesin maka akan dilanjutkan proses pada mesin selanjutnya.

4. Penjadwalan *job shop*

Klasifikasi penjadwalan dengan terdapat m mesin dimana setiap *job* memiliki rute produksi yang tidak tetap. *Job* dapat diproses lebih dari satu kali pada mesin yang sama. Kondisi ini yang sering disebut dengan *recirculation*.

5. Penjadwalan *open shop*

Klasifikasi penjadwalan dengan terdapat m mesin dimana setiap *job* memiliki urutan proses yang berbeda dan tidak ada batasan urutan produksi untuk setiap *job*, beberapa *job* memiliki rute sendiri dan yang lainnya tidak.

Pada prinsipnya, menurut Ginting (2009: 90) ada dua macam pendekatan penjadwalan, yaitu:

1. Penjadwalan secara *forward*

Penjadwalan *forward* adalah penjadwalan operasi dari saat mulai, bergerak searah dengan pergerakan waktu sampai seluruh operasi terjadwalkan. Penjadwalan ini digunakan bila yang ditentukan adalah saat mulai. *Output* dari penjadwalan adalah saat selesai. Keunggulan penjadwalan *forward* antara lain baik untuk antisipasi operasional tak terduga misalnya *breakdown machine*, penyisipan *job*, dan sebagainya sehingga cocok untuk penjadwalan dinamik. Sedangkan kelemahannya adalah kurang baik untuk mengantisipasi *due-date*.

2. Penjadwalan secara *backward*

Penjadwalan *backward* adalah penjadwalan operasi dari *due-date*, bergerak berlawanan arah dengan pergerakan waktu, sampai seluruh operasi terjadwalkan. Penjadwalan ini digunakan bila yang ditentukan adalah saat selesai. *Output* dari penjadwalan adalah saat mulai. Keunggulan penjadwalan *backward* antara lain baik cocok untuk mengantisipasi *due-date* dan tepat untuk meminimasi ongkos *tardiness*. Sedangkan kelemahannya adalah kurang cocok untuk antisipasi operasional tak terduga.

2.4 Metode Penjadwalan Produksi

Dalam menyelesaikan atau memecahkan suatu masalah penjadwalan, dibutuhkan suatu metode yang tepat. Secara umum, ada dua metode yang digunakan, yaitu dengan *linear programming* dan heuristik. Metode heuristik merupakan metode yang melakukan pendekatan suatu solusi optimal. Kelebihan metode heuristik adalah tahapan yang digunakan lebih sederhana dan efisien, serta menghasilkan hasil yang mendekati optimal. Metode heuristik dapat dilakukan dengan berbagai cara, salah satunya metode yang paling simpel membutuhkan pengurutan pekerjaan. Hal ini dapat dilakukan dengan *priority dispatching rule* (memprioritaskan pekerjaan).

Priority dispatching rule adalah aturan penjadwalan yang mengatur *job* dimana pada suatu *job* pada suatu mesin yang harus diproses terlebih dahulu berdasarkan prioritas-prioritas tertentu. Jadi pada saat suatu mesin telah selesai memproses satu *job* berdasarkan *priority dispatching rule* dipilih salah satu *job* yang memiliki prioritas tertinggi untuk selanjutnya diproses pada mesin tersebut (Pinedo, 2012: 376).

Dispatching rule memiliki beberapa aturan yaitu *Earliest Due Date (EDD)*, *Minimum Slack Time (MST)*, *Shortest Processing Time (SPT)*, *Longest Processing Time (LPT)* dan aturan lainnya. Pada sub-bab berikutnya akan dibahas mengenai pendekatan LPT.

2.4.1 Metode *Longest Processing Time*

Metode *longest processing time* merupakan metode penjadwalan yang memberikan prioritas tertinggi pada waktu penyelesaian *job* paling lama diselesaikan. Metode ini bertujuan untuk meminimalkan *makespan*. Algoritmanya adalah sebagai berikut (Baker, 1974):

1. Urutkan semua *job* berdasarkan waktu proses terpanjang (*longest processing time*).
2. Tempatkan *job* terawal dari hasil pengurutan pada mesin yang memiliki beban terkecil. Bila beban sama, pilih mesin secara acak. Ulangi hingga semua *job* habis.

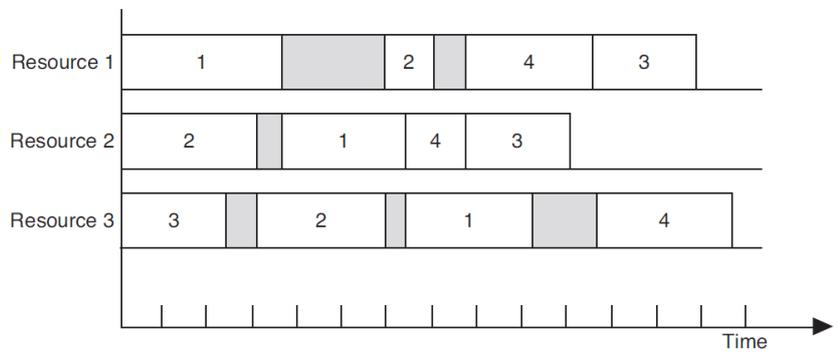
2.5 Gantt Chart

Menurut Pinedo (2012: 471), ada beberapa model/*interface* yang tersedia untuk mempermudah visualisasi dari penjadwalan secara detail antara lain:

- a. *Gantt Chart*: menunjukkan kapasitas yang tersedia dalam jangka pendek (hari atau minggu) ketika ada sejumlah pekerjaan/*job* (dua puluh atau tiga puluh).
- b. *Dispatch List*: menunjukkan daftar pekerjaan di urutan mana pekerjaan tersebut akan diproses pada suatu mesin, namun *interface* ini tidak memberikan pandangan yang baik dari jadwal yang relatif terhadap waktu.
- c. *Capacity Bucket*: menunjukkan angka/numerik pada setiap segmen waktu, utilitas setiap mesin dan baik untuk perencanaan jangka mediu atau panjang.
- d. *Throughput Diagram*: menunjukkan secara kumulatif dari waktu ke waktu mengenai jumlah total pesanan yang diterima, total yang dihasilkan dan total yang dikirimkan.

Model yang paling sederhana dan paling banyak dipakai adalah *gantt chart*. *Gantt chart* dipopulerkan penggunaannya oleh Henry Gantt pada tahun 1990an. Pada umumnya, *gantt chart* terdiri dari garis vertikal dengan sumbu y, dan garis horisontal dengan sumbu x. Seperti pada Gambar 2.2, sumbu y mewakili sumber daya tertentu, sedangkan sumbu x

menunjukkan skala waktu. *Gantt chart* tersebut juga menunjukkan kapan *job* akan mulai hingga selesai dan juga menunjukkan perkiraan waktu *idle* untuk setiap sumber daya.



Gambar 2.2 *Gantt Chart*

Sumber: Baker & Trietsch (2009: 3)

Dengan *gantt chart*, akan lebih mudah menemukan informasi tentang jadwal yang diberikan dengan menganalisis hubungan geometris. Selain itu, juga dapat mengatur ulang tugas atau *job* pada grafik untuk memperoleh informasi komparatif tentang jadwal alternatif. Dengan cara ini, *gantt chart* berfungsi sebagai bantuan untuk mengukur kinerja dan membandingkan jadwal serta untuk memvisualisasikan masalah sejak awal.